加微信：642945106 发送“ 赠送 ”领取赠送精品课程
发数字“ 2 ”获取众筹列表

02 | 方法论：没有任何文档，怎么才能快速了解接口的信息？

2020-02-03 陈磊

接口测试实战课 进入课程 >

讲述：陈磊
时长 19:22　大小 15.53M

你好，我是陈磊。

我相信在学习完上一节课后，你已经明白了接口测试是在测什么，我们为什么需要做接口测试。那么，当你面对一个接口测试任务的时候，你知道该如何开始吗？其实，任何事情从 0 到 1 都是一个门槛，你只要跨过这个门槛，后面就会一马平川。今天我就来告诉你，如何开始接口测试，让你面对一个项目不再束手无策，也不再面露难色。

说起接口测试，我想你并不陌生。作为一名测试工程师，尤其是做了多年业务测试的测试工程师，在开始接触接口测试时，无论开发工程师是否提供了接口文档，我相信你都会对下面几种场景似曾相识：

1. 开发工程师提交测试的项目，附带着一个几十页的 Word 文档，里面是一行一行的访问地址和路由，面对这样的 Word 文档，不知道如何开始验证；
2. 开发工程师在即时通讯工具上，甩给你有好几页的这么一个传输消息，里面有各种嵌套的参数，你不知道这些参数都是干什么用的；
3. 开发工程师口头告诉你需要测试的接口地址，然后就什么都没再多说，你问了他几句话后，他就借口说自己忙，不再理你，而你看到那个又长、又复杂的地址，束手无策。

难道，面对这些状况，测试工程师就没办法自己分析接口，完成测试吗？我现在告诉你，当然不是。

接下来，我就带你一起看看，一个理想的提测项目是什么样的，在实际工作中，绝大部分的提测项目又是什么样的，然后我们一起看看，如何一步一步解决一个不理想的提测项目。

一个理想的提测项目

一个理想的提测项目，在提测的过程中应该既包含前期参与的产品需求、原型设计，这些是由产品经理来提供的；也包含后端接口文档、代码单元测试脚本，这些是由开发工程师提来供的。这些都是你开展测试的必要输入内容，它们具体有这些作用：

产品需求。它描述了系统的业务逻辑，通过这个文档，你才能知道怎么来设计测试用例；

原型设计。它会更加直观地告诉你系统的使用逻辑，这对测试用例的设计、对系统的前期认知都是有辅助作用的。

接口文档。它详细地描述了后端接口的访问方式和参数说明，使用这个输入项才能开展接口测试用例的设计、测试脚本的准备和测试数据的构建。

单元测试脚本。它是保障提测质量的必要环节，是研发工程师自测的一个有效手段，可以保障提测项目的提测质量。

这些内容不限制 SUT（System Under Test，被测系统）的类型，SUT 既可以是一个手机 App，也可以是一个 Web 服务，甚至还可以是一个微服务接口。

所以，对于接口测试阶段来说，一个非常理想的接口测试，就是从完美的接口文档开始的。开发工程师在设计和开发接口的过程中，就在不断维护和更新接口文档，这其中包含了每一

个接口的访问方式、访问路由、输入参数含义、返回参数含义，以及一个完整的例子。

这种接口文档可能是以 Word 文档形式存在，也有可能是以类似 Swagger 这种工具形式存在。说到 Swagger，它是我推荐给你的一个接口文档的存在形式，是一个从代码生成的、以 Web 服务形式存在的接口文档，它可以伴随代码的变更同步变化，这就减少了很多开发工程师和测试工程师之间的沟通成本。

由 Swagger 生成的接口文档的例子如下图所示，从图中你可以看到它对接口的访问方式、访问路由、参数都有着详细的描述。

防止断更 请务必加
当发微信：17161436665

这样，当你拿到接口文档时，就可以快速使用各种工具或者代码来完成你的单个接口测试任务了。与此同时，你还可以通过一些参数设计、参数上下文传递，来完成接口的流程测试。

理想的情况很难发生

上面我说的是很理想的情况，现实却往往并不总如人意，我相信你也肯定遇见过下面这些情况：

> 根据产品经理的一句话需求，开发工程师便开始任意发挥，“所见系统即需求”的情况普遍存在，这就更别说后续的单元测试和接口文档了；

开发工程师从来不写单元测试脚本，提测项目质量无法保障，接口文档更无从谈起，你不知道如何开始完成接口测试；

你在拿到提测项目后，从部署测试环境到开始测试，一直都是摸黑前进，由于接口测试没有充分的输入条件，只能从 UI 层开始测试，结果导致交付质量大打折扣。

就像墨菲定律说的那样：可能发生的事情必将会发生。所以，上面我列举的接口测试难以推行的一些常见情况，都是你在工作中会碰见的问题。

那么，如果一个项目没有接口文档，我们就无法开始接口测试了吗？当然不是。

测试工程师的工作本质上是一个由表及里的工作，如果你还是每次工作都处在和终端用户使用行为几乎一致的流程上，那么只能说明你还不算一名合格的测试工程师。

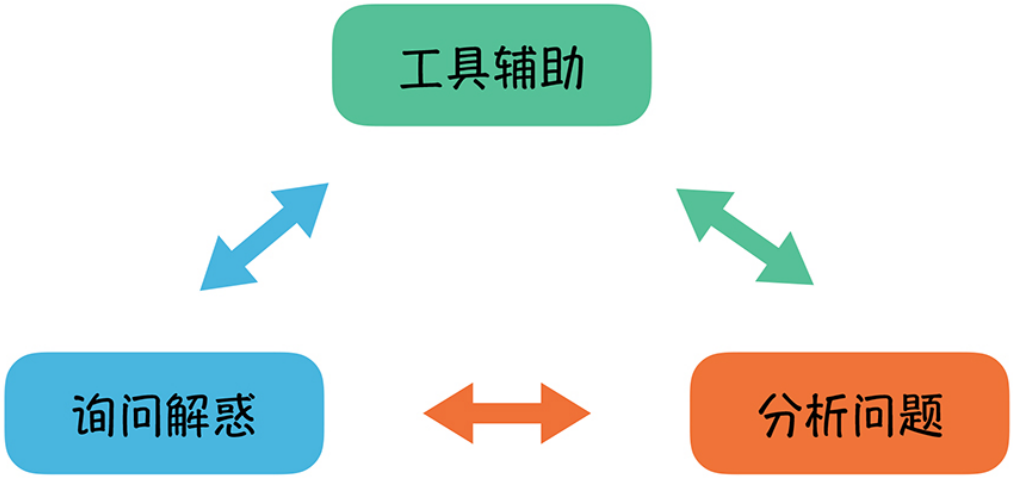
其实，无论开发工程师给我们的输入项是否包含了接口文档，我们都可以通过一些技术手段和工作方法，完成接口测试必要的输入项接口文档的创建。

那么这样的工作如何开始呢？下面我就来和你一起完成这样一项任务，教你如何开始第一个接口测试。

开始第一个接口测试

在拿到一个 SUT 环境的时候，你首先就要进行接口测试，这是因为单元测试不是由测试工程师来完成的，而是由开发工程师编写、并由持续集成系统自动完成执行的。

如果开发工程师没有给我们任何有价值的文档，那么要开始接口测试，你可以通过工具辅助、分析问题、询问解惑这三个步骤来完成。



具体的工作模式如上图所示：

1. 借助一些工具的辅助来完成接口分析；
2. 通过工具截获一些接口信息；
3. 通过分析接口的访问方式、参数等信息整理出一些问题，和研发工程师沟通这些问题，将一些不知道的参数含义、参数取值范围等问题问清楚。

拼课微信：1716143665

通过这三步的循环，你就可以完成对 SUT 系统接口信息的完善和维护，最终得到一份完整的、接口测试需要的输入—接口文档。

下面我会结合一个案例，带你看看这三步具体该如何进行。

工具辅助

当你第一次拿到一个被测项目，无论它是一个 App 服务还是一个 Web 服务，你都可以通过一些 HTTP 代理完成接口分析，这里我推荐你使用 Fiddler 这款工具。

> **注意：**
> Fiddler 既支持 Windows 操作系统，也支持 MacOS 操作系统，但是在 MacOs 上的版本并不好用，这是因为 Fiddler 使用了.Net 开发。如果你是一个 MacOS 深度用户，那么我推荐给你两款工具，一款是 Charles，另外一款是 mitmproxy。其中，Charles 是

> 商业软件，mitmproxy 是开源软件，但是 Charles 使用起来更简单，mitmproxy 的使用则稍微复杂一些，你可以依据自己的喜好来选择。

简单来说，Fiddler（在这里下载）是一个 HTTP 的调试代理，也就是一个 HTTP 协议的抓包工具，运行时会在本地建立一个代理服务，默认地址为 127.0.0.1:8888。当浏览器和被访问的服务之间发生交互的时候，Request（请求）和 Response（响应）都会经由 Fiddler 代理，这样就可以截获下全部的访问信息流了。

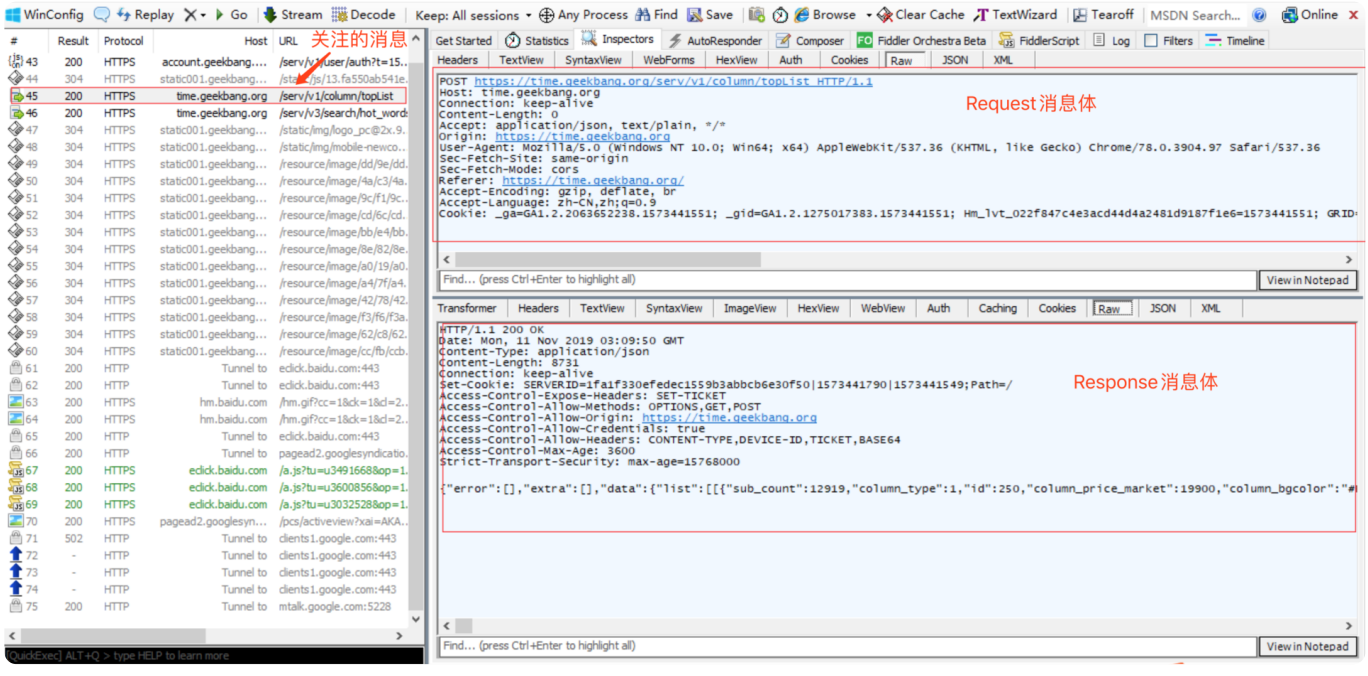
在接下来的内容中，我会带你使用 Fiddler 来完成接口的分析任务，**但是，我并不打算手把手的把 Fiddler 这个工具的使用细节讲述给你，因为除了 Fiddler，其实还有很多工具可以完成对应的任务，每一种工具又都有其具体的使用方法，这里我真正想告诉你的，是通过具体的方法来解决问题的思维。**

分析问题

在解决问题前，我们首先要分析问题，这也是我们开始接口测试的第二步。

下面我们一起来使用 Fiddler，分析一下极客时间 Web 端首页（极客时间 Web 端首页地址 https://time.geekbang.org）。

首先，打开 Fiddler，然后启动浏览器访问极客时间 Web 端首页，我们可以看到 Fiddler 截获了很多消息，找到刚刚输入的 https://time.geekbang.org，如下图所示，在右侧上方 Inspectors 的标签页下，你就可以看到 Request 请求的内容和 Response 请求的内容了。



为方便你查看，我将 Request 的消息体复制出来如下：

```
POST https://time.geekbang.org/serv/v1/column/topList HTTP/1.1
Host: time.geekbang.org
Connection: keep-alive
Content-Length: 0
Accept: application/json, text/plain, /
Origin: https://time.geekbang.org
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTM
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Referer: https://time.geekbang.org/
Accept-Encoding: gzip, deflate, br
Accept-Language: zh-CN,zh;q=0.9
Cookie: _ga=GA1.2.2063652238.1573441551; _gid=GA1.2.1275017383.1573441551; Hm_
```

从这段消息体中我们可以获知，它的访问方式是 POST，访问的 URI 是“https://time.geekbang.org/serv/v1/column/topList”。这里面的具体属性内容，你可以自行查看，但是，我希望你重点关注如下这几个属性：

HOST，它表示指定访问的服务器域名；

Connection 的值为 keep-alive，这表示需要持久连接；

Accept，它表示客户端可以接受的内容类型为 application/json, text/plain, /；

User-Agent，它说明请求是从什么浏览器发出去的；

Sec-Fetch-Site 和 Sec-Fetch-Mode，它们是 JS 中对跨域的一些设置；

Accept-Encoding 设置为 gzip、deflate、br，这表示可以支持的 Web 服务器返回内容压缩编码类型；

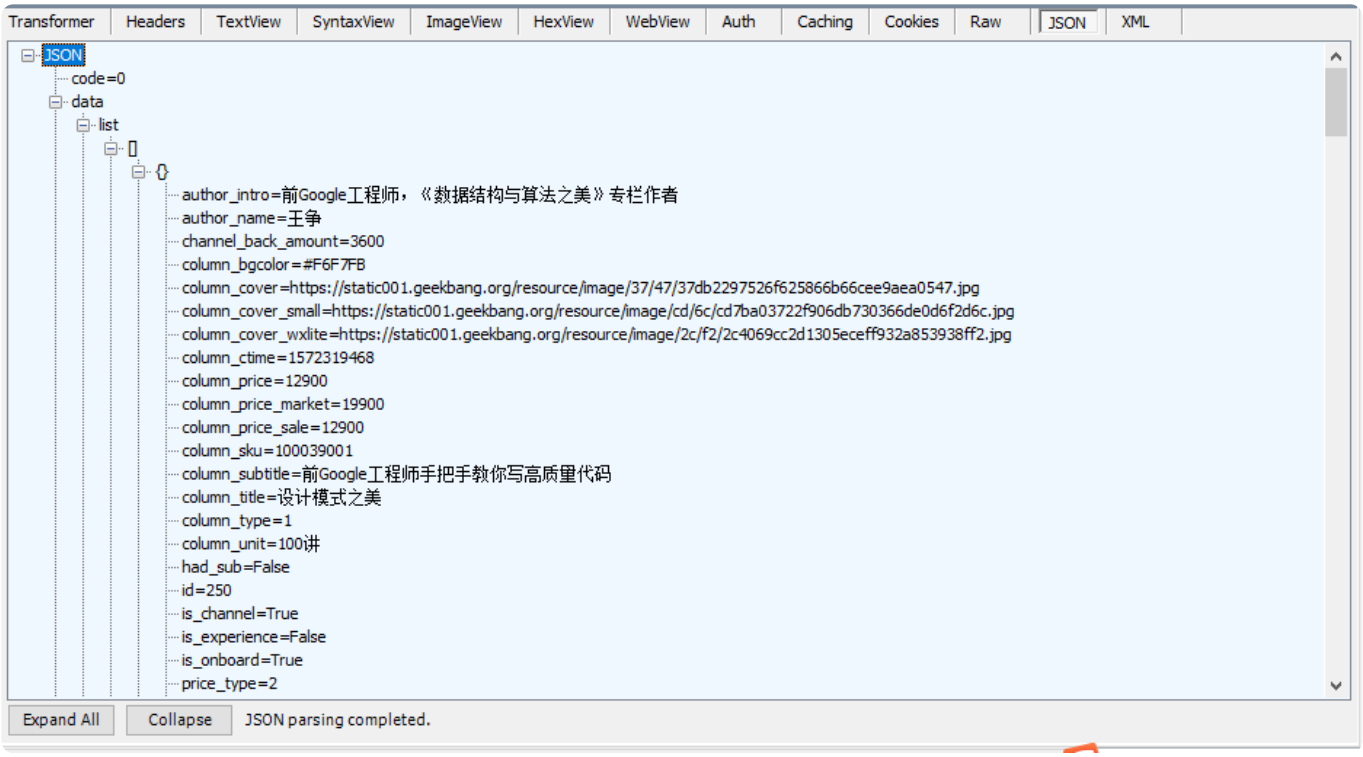
Accept-Language，它表示接受的语言。

这其中的 Cookie 的内容，是你需要特别非常关心的，因为 Cookie 中传递的参数很多都是用来确认用户身份、鉴定角色权限等需要的参数。这样，通过上面的分析你就可以自己绘制如下的表格，这里我也给出了一张表格。

访问路由	serv/v1/column/topList	
HTTP协议版本	V1.1	
访问方式	POST	
origin	https://time.geekbang.org	
Headers	Host	time.geekbang.org
	Connection	keep-alive
	Content-Length	0
	Accept	application/json, text/plain, */*
	Origin	https://time.geekbang.org
	User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.97 Safari/537.36
	Sec-Fetch-Site	same-origin
	Sec-Fetch-Mode	cors
	Referer	https://time.geekbang.org/
	Accept-Encoding	gzip, deflate, br
	Accept-Language	zh-CN,zh;q=0.9
Cookies	_ga	GA1.2.2063652238.1573441551
	_gid	GA1.2.1275017383.1573441551
	Hm_lvt_022f847c4e3acd44d4	1573441551
	GRID	b0a2570-01c8b13-90b002b-568dc07
	MEIQIA_VISIT_ID	1TS7HWSYajeZyS29IqNNyvW9cyY
	MEIQIA_TRACK_ID	1TS7HZW4C6OWaPSu5VGIj7uN4pM
	SERVERID	1fa1f330efedec1559b3abbcb6e30f50|1573441580|1573441549
	_gat	1
	Hm_lpvt_022f847c4e3acd44d4	1573441790
Body	无	

在这个表格中，被标注为白色背景的部分，是这次访问的基本信息；被标注为黄色背景的部分，是访问的头信息，同时也是我们已知的内容；被标注为红色背景的部分，就是 Cookie 信息，是我们未知的内容。同时你可以看到，本次消息访问的 body 是空的，是没有内容的。

接下来我们再看看这个请求的 Response 信息，由于返回的消息比较长，我在这里就不贴出来了，但是通过下面这个图你可以看出，这次返回的主体是一个很长的 JSON，这里面包含了各个专栏或者课程的信息。



这些返回值包含了很多参数，你也需要关注这些参数，因为很多时候，一个接口的返回值会是另外一个接口的入参，也就是我常说的串联业务逻辑上下文的参数。

现在，在你绘制的接口信息表中还有一些未知的 Cookies 内容，就如我前面说的，Cookie 内容是完成接口测试必须要模拟并传递的一些信息，因此，我们必须要尽可能完善它，使它成为接口测试的必要输入条件之一。

这样，拿着这张接口信息表，我们就进入了第三步，询问解惑。

询问解惑

对于本次访问的 Cookies 的参数，从参数语义上来说，我们无法知道这些参数是用来干什么的，他又起到了什么作用，作为测试人员，我们也无法只靠自己知道每一项具体的含义、表示的内容以及参数的作用。

因此，我们需要拿着上面的那张表格，找到对应的开发工程师，去问清楚表格中标红部分的参数。

针对每一个参数，你都要从下面的几点详细询问，并保证你已经真的理解了这些内容。那么，都询问些什么呢？我认为主要有三点。

1. **参数的含义以及来源。**你要搞清楚每一个参数的含义，也就是这个参数对应的实际自然语言的名字，通过记录每一个参数的中文语义，也会让更你容易记住这个函数是干什么的。同时，你也要知道这个参数的赋值是从哪里来的，是从其他页面的返回值中得到的？还是 JS 生成的？如果是其他页面或者接口返回的，那么，是哪一个接口返回的哪个字段？这样，当你开始做接口测试的时候，你就知道去哪里拿到这个参数的赋值了。如果是另一个接口的返回字段，那么，你还需要维护一份返回该参数接口的接口信息文档，以便于自己下一次创建对应的参数，如果不可以创建，那么你就要知道这个参数的生成规则，也要知道如何手动构造它。
2. **参数的作用域。**参数的作用域指的是这个参数在这个接口中是做什么用的，它在哪一个访问周期里是一直存在的，它是否导致了业务逻辑分支等。比如说，这个参数是用来验证用户权限吗？它的验证算法是什么？之所以要搞清楚这些内容，是为了你在做接口测试的时候，可以设计更小的参数组合来覆盖更多的业务逻辑，这是测试用例去除冗余的一个很好的方法。
3. **返回值的含义。**针对上面一大串的返回 JSON，你要搞清楚在返回值中，每一个 JSON 的 Key 所对应的含义，这样，当你需要和这个接口产生交互的时候，就可以快速地拿到对应参数的含义，完成业务逻辑上下文的参数串联了。

总地来说，Request 的全部参数和 Response 的全部参数对于接口测试来说，都是必要的输入项，因此我们有必要花费很多精力完善并且留存它们。

OK，到现在，你已经借助工具、通过分析问题明确了未知参数，也通过询问解决了未知参数的中文含义、作用域，以及对应返回参数的中文含义，现在，即使面对没有接口文档的提测项目，你也能收集明确的、足够的信息了。

那么接下来，你就可以利用这些信息，完成业务逻辑的接口测试了。

多个接口串行分析

在质量保障过程中，测试的主要任务，是保障 SUT 的业务逻辑正确性，而单一接口的测试却很难完成一个业务逻辑，所以，在大部分的测试场景中，我们都需要串行多个接口，才能完成一个完整的业务逻辑。

然而，即使我们按照上述三个步骤完成了全部单个接口的分析，也并不能马上开始进行接口测试。这是因为，一个测试的业务逻辑是由多个接口的串行完成的，而多个接口的串行逻辑是由业务逻辑规定的，因此，多个接口之间并不是随意组合的，而是按照业务逻辑、通过数据传递来完成的。

这其实就和拼图游戏一样，我们有一堆拼图碎片，很多拼图碎片都可以连接到一起，并不会有明显的不适合，但是，依据拼图的最终图形，这些拼图碎片就是不能放到一起。你要想把拼图完成，就不仅要考虑各个拼图碎片是不是可以链接到一起，还要考虑这些碎片放到一起后是不是对原来图形的正确拼接。

那么，你前面整理好的、各个单一接口的信息表，就是拼图游戏里的一个拼图碎片，业务逻辑就是拼图组成的最终图形，而其中的参数，就是拼图碎片的缺口和每一个碎片上的图形。

所以，要想使用接口测试完成业务逻辑，你就要制作一个流程中所有接口的接口信息表，同时，还要理清每一个流程的数据流程，数据流程驱动了业务流处理，这样，才能开始业务逻辑的接口测试。

关于怎么完成一个业务逻辑的接口测试，我会在下一节课中详细的解释给你。

总结

好了，今天的内容就到这里了，不知道你是不是已经掌握了分析一个接口并整理接口信息表的方法呢？现在我来总结一下。

在今天的课程中，我带你一起分析了一个理想的提测项目都包含哪些要素，但是，这种乌托邦式的提测项目在现实中却很难遇见。所以，我给你推荐了“三步走”的方法，你可以通过工具辅助、分析问题和询问解惑这三步来建立接口信息表，建立和维护自己的接口知识库。我也通过拼图游戏的例子，告诉你接口测试的业务逻辑验证思路。

随着被测系统的不断迭代，当你不断地按照这个步骤积累起自己的接口知识库时，你就会拥有自己负责的业务线的业务知识积累，变成这条被测业务线的业务专家；同时你也会拥有每一个接口的输入和输出全部参数、每一个业务逻辑的数据流、每一个驱动业务分支逻辑的参数条件，你也会成为接口测试的技术专家，在你自己的工作中变成一个不可或缺的质量保障者。

在一个团队中，有很多时候，测试工程师和开发工程师之间就是一个矛盾的共同体，既相互依存又不可分割，在团队中缺一不可，要想摆脱“测试非技术岗位”的帽子，就需要测试人员有自己的技术功底和技术素养。

无论你在工作中，遇到的是能很好相互支持的开发工程师，还是不好合作的开发工程师，你都要保持自己的技术能力，尽最大努力完成自己能够完成的所有事情，只有这样，你才能提高自己在团队中的话语权。

思考题

最后，我想请你来思考一下，你能用本节课讲述的 Fiddler 还有接口测试分析方法，完成一个你自己项目的老旧接口的分析吗？

我是陈磊，欢迎你在留言区留言分享你的观点，如果这篇文章让你有新的启发，也欢迎你把文章分享给你的朋友，我们一起沟通探讨。

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇　01 | 基础：跳出细节看全局，接口测试到底是在做什么？

下一篇　03 | 思维方式：用一个案例彻底理解接口测试的关键逻辑

精选留言 (28)

写留言

胖虫子
2020-02-04

刚来个要求，把app所有接口压测一遍，半天时间，有的时候最难的不是工作，而是遇到这样的傻逼领导

1 10

SugarZh
2020-02-04

类似于Fiddler或Charles这类的抓包分析工具，并非所有公司都允许使用；例如：银行等金融性的公司，此类工具都是不在白名单的；所以个人觉得，还是要尽可能去推动完善项目流程规范，只有在一个完善的规范之下，才能更好的协调和改善工作效能，提高测试质量

展开

作者回复: 您说的没错，谢谢您的留言，我在招商银行信用卡中心的时候也有过类似问题，当时我解决问题的方法是自己写了一个简单的proxy，这就顺利逃脱了冗长的流程申请。

1 4

小尘埃
2020-02-04

工作中都是用jmeter做接口测试，但是接口脚本维护却比较麻烦。人手一份。

作者回复: jmeter做接口测试其实没有问题，问题出在了script的管理管理，建议使用git做统一管理。谢谢您的支持

4

苦行僧
2020-02-03

我们公司是部门强制，开发必须和测试结对完成接口api测试用例的设计，包括设置各种参数

作者回复: 谢谢您，很好的一种模式，本来测试和开发就是一家，结对也是有益的

4

happychap
2020-02-04

老师您好，我有一个疑问。就是：我们如果把接口测试放在ui界面出来之后再做，虽然它确实会比ui自动化测试用例更稳定，但接口测试的介入的时间点会不会太晚了一些呢？毕竟这个阶段所有开发工作都已经完成了，修复软件bug的成本已经是比较高的了。

展开

3 3

Angela
2020-02-04

公司项目的单接口、串联接口都已经在做 但是好多时候接口已经测试完 然后开发改了部分接口，导致接口再次自动化回归的时候就又有问题了 这样很被动，请问下如何解决此类问题

作者回复: 变更频繁这个问题目前只能在项目内部解决，服务端工程师应该设计接口，所有相关人评审后就不能变更，要变更需要有人承担成本。这个可以通过引入敏捷模式来进行规避，您可以关注一下对应内容。

2 3

从一
2020-02-06

对于一个新项目，还是要项目经理，测试经理推动开发人员维护完善的接口文档，从项目流程保证和完善；
对于一个存量项目，可以采用老师介绍的三部曲，工具辅助，分析问题，询问解惑。老师以http(s)协议给出了完整的分析思路。也期待接下来的课程。

展开

作者回复: 谢谢，您说的特别好

2

彭序猿
2020-02-09

咨询一下，接口知识库的表，有什么软件推荐吗？

展开

作者回复: 目前没有软件推荐，我常用excel，不知道是不是可以在confluence上建立一个模版。谢谢

1

肥肥肥猫
2020-02-04

1、没有文档，可以开始做一些接口，让项目经理、开发或者产品经理尝到甜头，然后还是推动规范化，拜托保姆式测试~
2、期待磊哥的二叉树当初的设计思路

作者回复: 二叉树的aat不会讲哦！谢谢您的支持，你说的方法也是对的先来的做一点接口测试让团队boss看到好处给予支持

1

可可
2020-02-04

对于websocket接口，如何探测他的各种接口呢？好像没有接口文档不好搞啊

作者回复: websocket确实不好处理后续我们也回了的websocket希望有所启发！谢谢支持

1

包Joyce
2020-02-03

接口测试什么时候介入比较合适?
敏捷开发模式下，如何进行接口测试呢?

作者回复: 接口测试应该早介入，在接口已经开发完最好就介入，但是前期分析开始借助一些工具就开始了，在敏捷模式下，接口测试确实更有施展的余地，还可以借助契约测试完成解耦和api测试

1

summer
2020-02-03

感谢答疑解惑，持续关注后续更新干货

展开

作者回复: 谢谢您，欢迎和我一起来学习，我们互相鼓励！

1 1

7F

2020-02-09

有时候工作内容不仅仅涉及到自家的产品，可能还要去窥视别人家的APP，有时候别人家的APP抓包要么抓不到，要么就是请求参数和返回参数做了类似加密的方式，好难啊。

作者回复: 谢谢，技术在坚持，一些三方应用拿不到param可以直接询问三方合作伙伴

Daiver

2020-02-07

如果要整理出接口的信息表，那会花太多时间。

展开

作者回复: 技术债要慢慢还，这些技术债是我们之前遗留下来的，时间越久债务越多。但是不提倡一次还清，要随着时间的积累慢慢减少。谢谢

凌波微步

2020-02-07

平时工作中都是摸黑前行，F12查看接口信息去设计接口测试脚本，但实际中会发现一些无意义的请求参数或者返回值，类似这样的冗余参数属于接口设计不规范问题，也应该正面提出问题。避免再犯

作者回复: 谢谢经验分享！

花菜君大饼

2020-02-06

请问会讲一下dubbo接口测试嘛

展开

作者回复: 谢谢，不会涉及的这部分内容了

lulu

2020-02-05

fiddler抓包web端https，总是不成功。如何配置自己配置代理，求大佬指点一二

作者回复: fiddler提供了证书，你可以具体查询一下如何使用。

VeryYoung

2020-02-05

前期研发也是不提供完整的文档，后面就强制㊣怕

展开

作者回复: 谢谢留言，靠自己慢慢积累也会有所得

月亮和六便士

2020-02-05

老师，我想问一下，一般接口测试的时候，ui一般都没开发好，这种情况下，怎么通过代理工具拿到接口信息?

展开

作者回复: 代理拿到接口是说一个历史项目，一个全新项目在没有ui的时候还是要通过开发给出或者自己查看源代码。谢谢您

TC123

2020-02-05

chrome按F12就能拿到接口信息，但有时候我们想拿到后端接口，这个才是难点

作者回复: 您的方法我也常用，这也是我说的工具不重要，思维才是最重要的原因